Crafting Large Language Models for Enhanced Interpretability

ICML 2024 Mechanistic Interpretability Workshop

Authors: Chung-En Sun, Tuomas Oikarinen, Tsui-Wei Weng

Main Problem

Main Problem

Lack of **interpretability** in current LLM models

➔ hard to debug or improve

➔ e.g. GPT, BERT

To solve this

Concept Bottleneck Large Language Model (CB-LLM)

Objective

Offer intrinsic interpretability while maintaining the high performance of black-box models

Existing Approaches to Solve This Problem

Approach	**Post-hoc Neuron Analysis**	**LLMs as Post-hoc Explainers**	**Concept Bottleneck Models (CBM) for Images**	**CLIP-Dissect (Label-free CBM for Images)**	**Contrastive Learning for Sentence Embeddings**
Purpose	To understand the inner workings (neuron activations) of black-box models.	Uses a more powerful LLM (e.g., GPT-4) to generate explanations for neurons in smaller models (GPT-2).	Introduces a "concept bottleneck layer" where neurons explicitly learn human-interpretable concepts.	Uses CLIP to automatically identify concepts without human annotations.	Techniques like SimCSE leverage entailment pairs to train embedding models.
Limitation	1.Often fails to align with actual neuron functions 2.Lack clear guidance for model debugging.	1.May oversimplify neuron behavior, explanations do not fully capture model internals 2.High computational cost	1.Requires human-annotated concept labels 2.Build a CBM from scratch is computationally expensive.	1.Still constrained to visual models 2.does not fully eliminate the need for labeled concepts.	1.Does not directly improve model interpretability 2.focuses more on embedding quality rather than structured reasoning.

Key Contributions

1

Concept Bottleneck Layer(CBL): Ensures that LLMs make predictions based on interpretable concepts rather than opaque internal features

Input: fixed-size embedding from pretrained LM;
Output: activation vector represents how strongly each learned concept is present in the input text.

2

Automatic Concept Scoring (ACS): Automatically identifies concepts in text without requiring human annotation

Input: Text;
Output: Concept relevance scores

3

Automatic Concept Correction (ACC): Uses GPT-4 to refine and correct the noisy concepts generated by ACS

Input: Initial list of generated concepts;
Output: More accurate and meaningful concept explanations

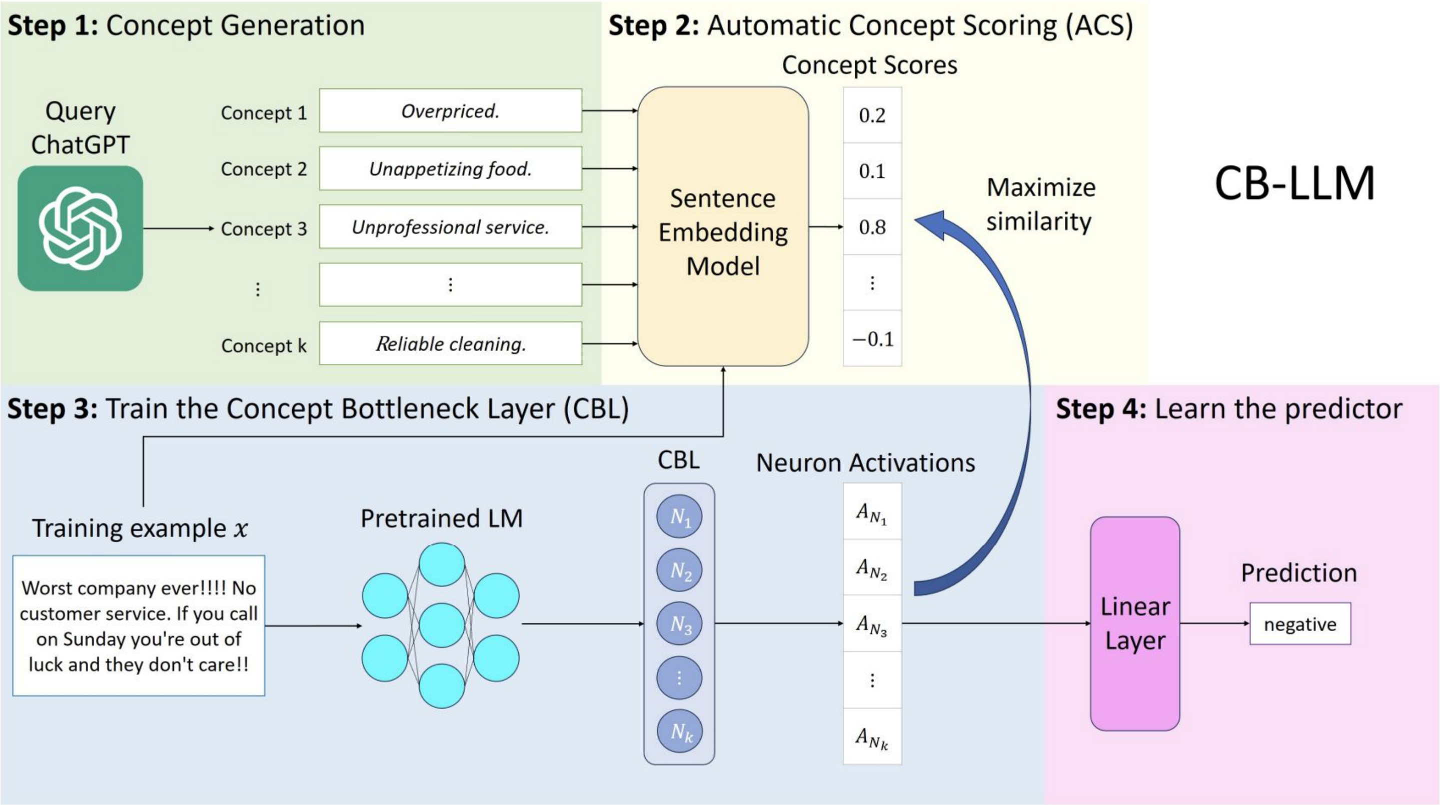


Figure 1. The overview of our CB-LLM.

CB-LLM

Key Equations

ACS:

$$S_c(x) = \left[\mathcal{E}(c_1)\cdot\mathcal{E}(x), \mathcal{E}(c_2)\cdot\mathcal{E}(x), \ldots, \mathcal{E}(c_k)\cdot\mathcal{E}(x)\right]^\top$$

Train CBL:

$$\max_{\theta_1,\theta_2} \frac{1}{|\mathcal{D}|}\sum_{x\in\mathcal{D}} \mathrm{Sim}\left(f_{\mathrm{CBL}}\left(f_{\mathrm{LM}}(x;\theta_1);\theta_2\right), S_c(x)\right)$$

Learn predictors:

$$\min_{W,b} \frac{1}{|\mathcal{D}|}\sum_{x,y\in\mathcal{D}} \mathcal{L}_{\mathrm{CE}}\left(WA_N^\dagger(x)+b, y\right)+\lambda R(W)$$

Experiment Setup

- **Datasets**: SST2, Yelp Polarity(YelpP, 560,000 training samples), AGnews, DBpedia
- **Backbones**: RoBERTa-base and GPT-2 base
- **Baseline**: fine-tuned RoBERTa-base (standard black-box model)
- **Evaluation Criteria**: Accuracy, Efficiency, Faithfulness
- **Goal**: High accuracy, Minimal additional cost, High interpretability

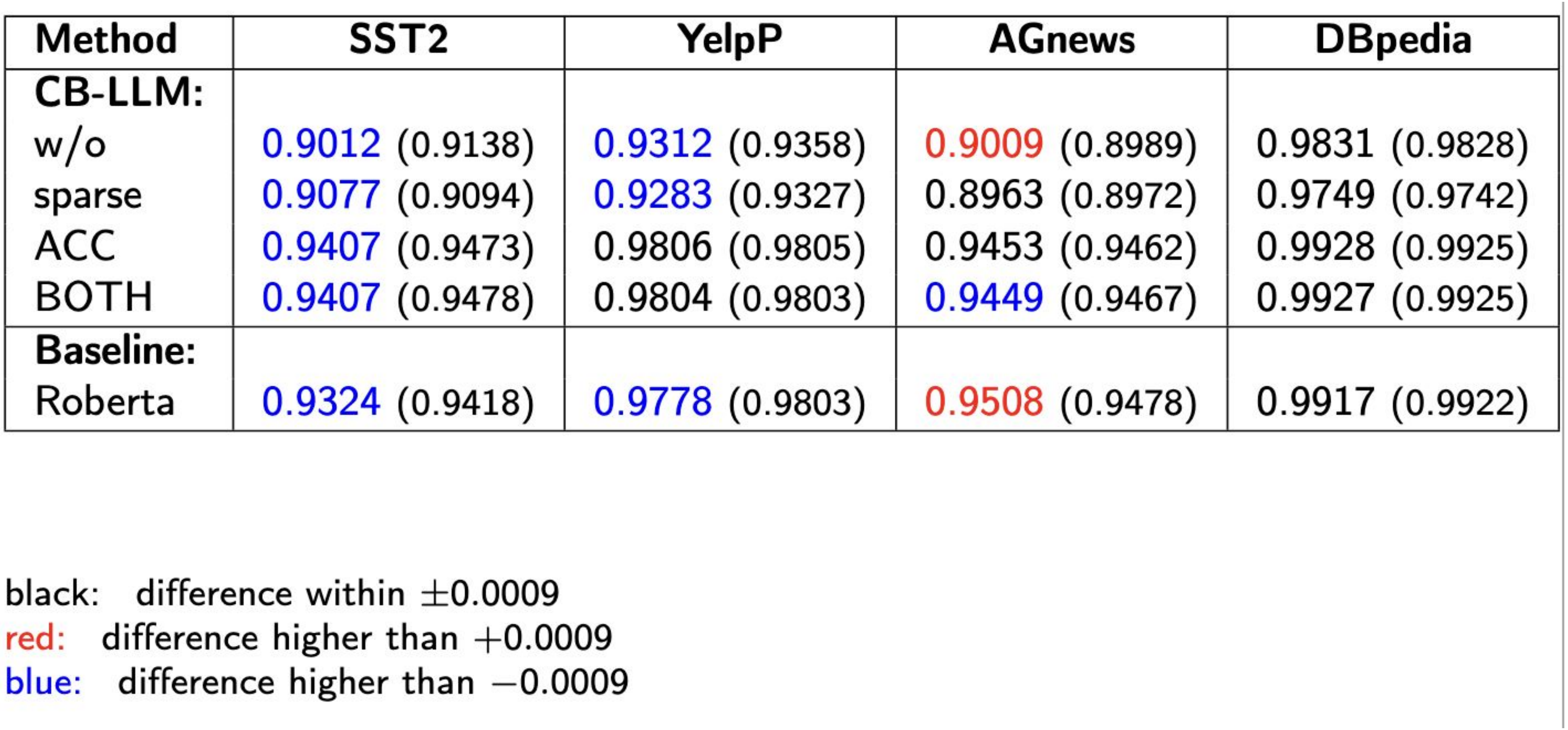
Experiment Result

- **Accuracy**: CB-LLM achieves competitive accuracy with black-box models for all datasets, demonstrating minimal performance loss; Automatic Concept Correction (ACC) consistently improves the accuracy of CB-LLM(See Table 1).

- **Efficiency**: CB-LLM balances performance and computational efficiency, introducing only a modest time overhead compared to black-box models(See Table 2, time cost).

- **Faithfulness**: Results of human evaluation suggests that the explanations generated by CB-LLM w/ ACC are better than randomly selected explanations.(See Table 3, Figure 3-5).

Replication: Accuracy



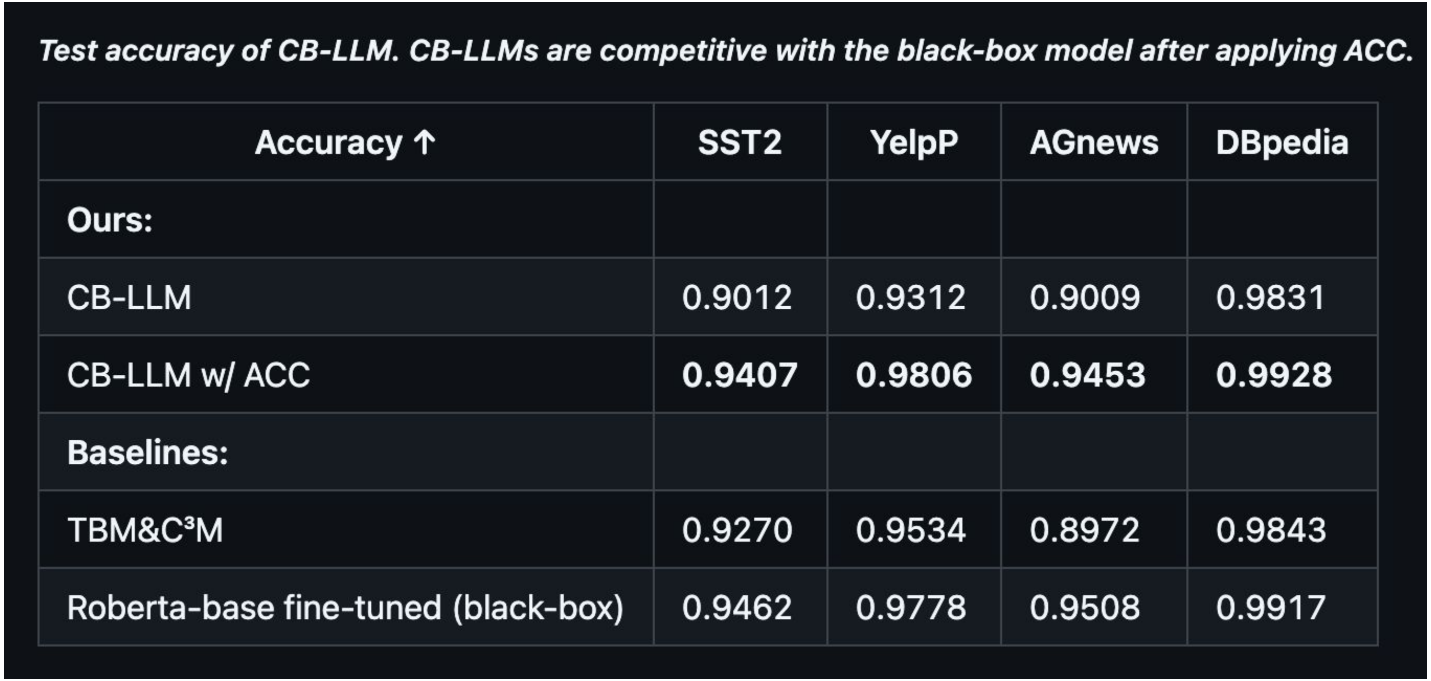
Method	SST2	YelpP	AGnews	DBpedia
CB-LLM:				
w/o	0.9012 (0.9138)	0.9312 (0.9358)	0.9009 (0.8989)	0.9831 (0.9828)
sparse	0.9077 (0.9094)	0.9283 (0.9327)	0.8963 (0.8972)	0.9749 (0.9742)
ACC	0.9407 (0.9473)	0.9806 (0.9805)	0.9453 (0.9462)	0.9928 (0.9925)
BOTH	0.9407 (0.9478)	0.9804 (0.9803)	0.9449 (0.9467)	0.9927 (0.9925)
Baseline:				
Roberta	0.9324 (0.9418)	0.9778 (0.9803)	0.9508 (0.9478)	0.9917 (0.9922)

black: difference within ±0.0009

red: difference higher than +0.0009

blue: difference higher than −0.0009

Replication: Accuracy

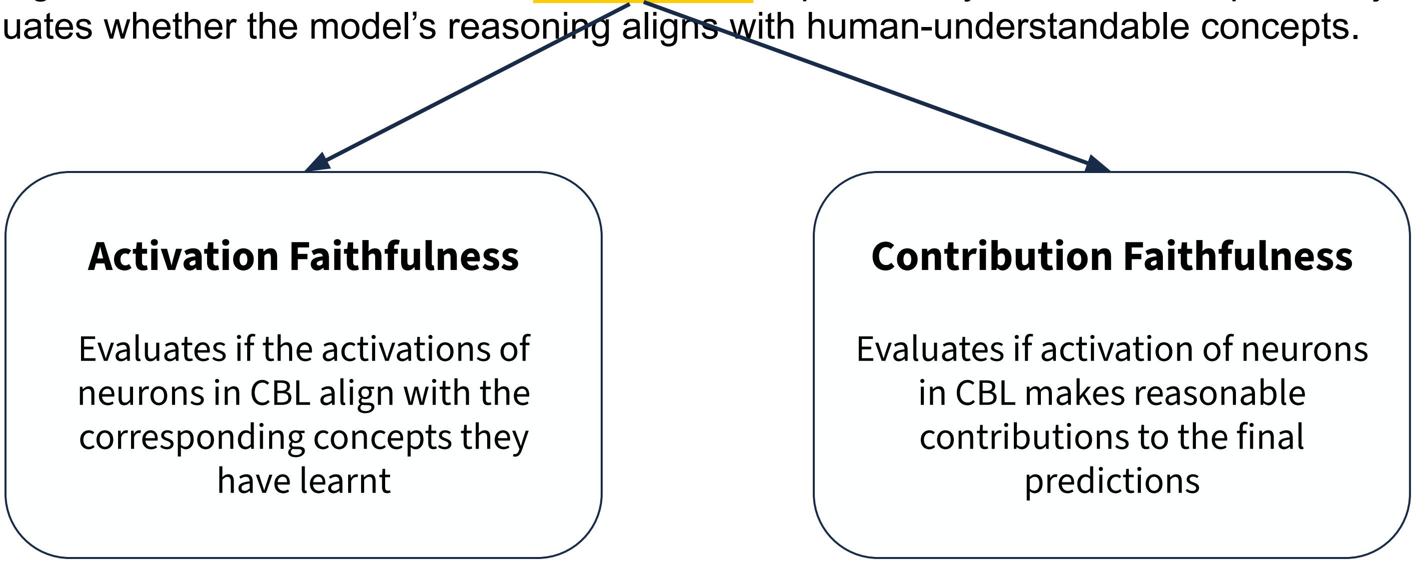


Test accuracy of CB-LLM. CB-LLMs are competitive with the black-box model after applying ACC.

Accuracy ↑	SST2	YelpP	AGnews	DBpedia
Ours:				
CB-LLM	0.9012	0.9312	0.9009	0.9831
CB-LLM w/ ACC	**0.9407**	**0.9806**	**0.9453**	**0.9928**
Baselines:				
TBM&C^{3}M	0.9270	0.9534	0.8972	0.9843
Roberta-base fine-tuned (black-box)	0.9462	0.9778	0.9508	0.9917

Key Observations: Interpretability

Among the three evaluation criteria, **Faithfulness** is particularly related to interpretability, which evaluates whether the model’s reasoning aligns with human-understandable concepts.



Activation Faithfulness

Evaluates if the activations of neurons in CBL align with the corresponding concepts they have learnt

Contribution Faithfulness

Evaluates if activation of neurons in CBL makes reasonable contributions to the final predictions

Experiment method to prove interpretability

1. **Human Evaluation for Activation Faithfulness**
    a. Task: Workers need to provide a rating ranging from 1 (strongly disagree) to 5 (strongly agree) based on the agreement observed between the neuron concept and the top k highly activated samples.
    b. Human Evaluation Result(compare CB-LLM with random baseline):
        i. CB-LLMs w/ ACC constantly achieve higher ratings than the random baseline; the neurons in our CB-LLMs w/ ACC are more interpretable than the neurons with random activations.

2. **Human Evaluation for Contribution Faithfulness**
    a. Task: Workers will be presented with explanations from two models for a text sample, and they need to compare which model's explanations are better.. The explanations are generated by showing the top r neuron concepts with the highest contribution to the prediction.
    b. Human Evaluation Result(compare CB-LLM with random baseline):
        i. Workers consistently express a preference for our CB-LLM w/ ACC over the random baseline; CB-LLM w/ ACC are better than randomly selected explanations

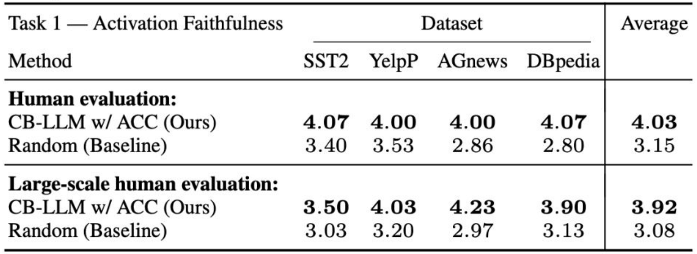
Experiment method to prove interpretability

3. **Ablation Study on ACC and Sparsity for Contribution Faithfulness**
    a. Evaluation: Impact of ACC on faithfulness and interpretability by comparing CB-LLM with and without ACC, and including a sparse final layer.
    b. Result:
        i. CB-LLM with ACC consistently outperforms the baseline in both Activation and Contribution Faithfulness.
        ii. Sparse final layer may only offer marginal help for the interpretability since the workers exhibit little preference after the application of a sparse final layer.

4. **Case Study: Concept Unlearning**
    a. What is Concept Unlearning: Force the model to forget a certain concept by manually deactivating a specific neuron in CBL or removing all weights connected to the neuron in the final linear layer.
    b. Evaluation: Shows how unlearning certain concepts (e.g.,"overpriced") impacts model predictions.
    c. Interpretability Proof: CB_LLM has great potential to facilitate human interventions such as conceptual unlearning to enhance fairness and flexibility.

Tables regarding interpretability

Table 3. The human evaluation results for task 1 — Activation Faithfulness. Workers give higher ratings to our CB-LLM w/ ACC, suggesting that the neurons in our CB-LLM w/ ACC are more interpretable than the neurons with random activations.



Task 1 — Activation Faithfulness	Dataset				Average
Method	SST2	YelpP	AGnews	DBpedia	
Human evaluation:					
CB-LLM w/ ACC (Ours)	**4.07**	**4.00**	**4.00**	**4.07**	**4.03**
Random (Baseline)	3.40	3.53	2.86	2.80	3.15
Large-scale human evaluation:					
CB-LLM w/ ACC (Ours)	**3.50**	**4.03**	**4.23**	**3.90**	**3.92**
Random (Baseline)	3.03	3.20	2.97	3.13	3.08

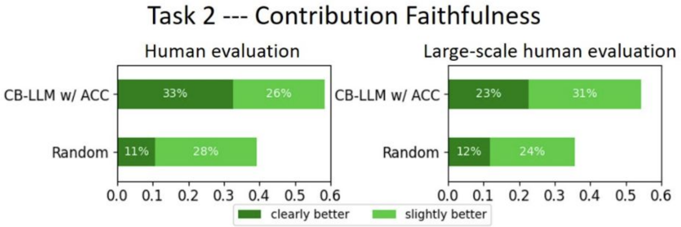


Figure 3. The human evaluation results for task 2 — Contribution Faithfulness. Workers prefer the explanations generated by CB-LLM w/ ACC more than the random explanations.

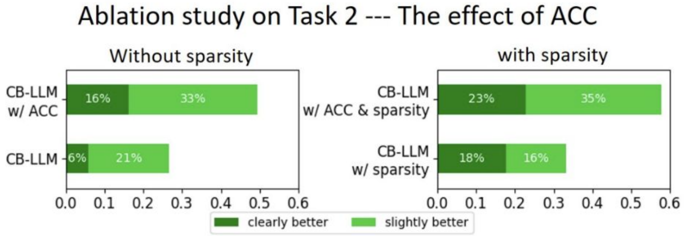


Figure 4. Ablation study on Automatic Concept Correction (ACC). Workers favor the explanations provided by the CB-LLMs with ACC.

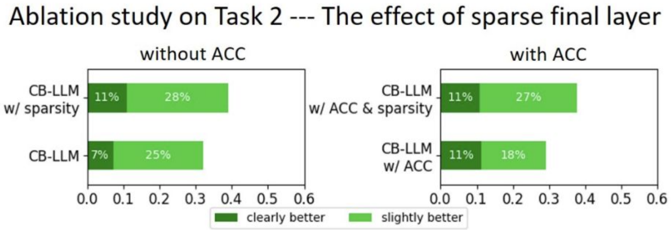
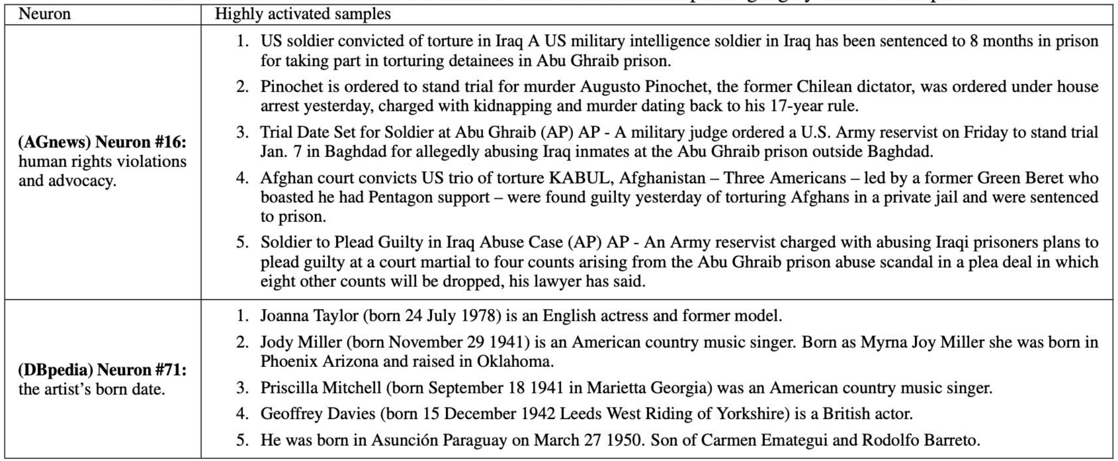


Figure 5. Ablation study on the sparsity. Workers demonstrate only a marginal preference for explanations provided by the CB-LLMs with a sparse final layer.

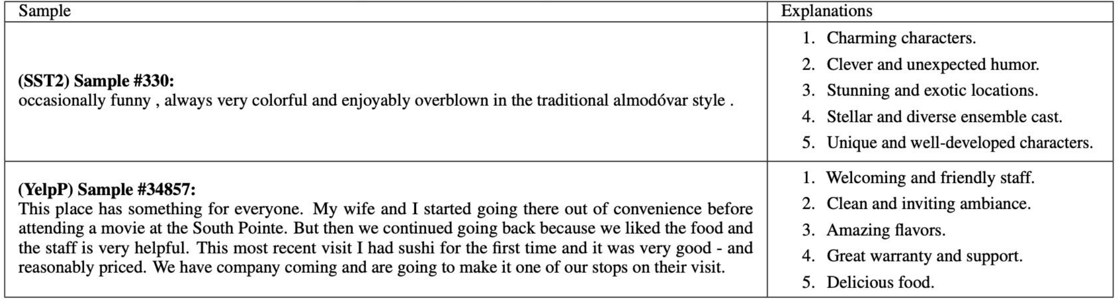
Tables regarding interpretability

Table 4. The two neurons from CB-LLM w/ ACC and their corresponding highly activated samples.



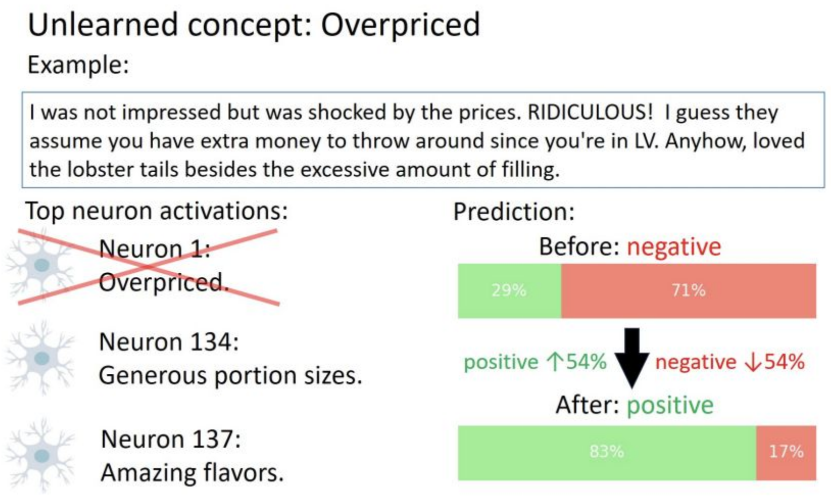
Neuron	Highly activated samples
(AGnews) Neuron #16: human rights violations and advocacy.	1. US soldier convicted of torture in Iraq A US military intelligence soldier in Iraq has been sentenced to 8 months in prison for taking part in torturing detainees in Abu Ghraib prison. 2. Pinochet is ordered to stand trial for murder Augusto Pinochet, the former Chilean dictator, was ordered under house arrest yesterday, charged with kidnapping and murder dating back to his 17-year rule. 3. Trial Date Set for Soldier at Abu Ghraib (AP) AP - A military judge ordered a U.S. Army reservist on Friday to stand trial Jan. 7 in Baghdad for allegedly abusing Iraq inmates at the Abu Ghraib prison outside Baghdad. 4. Afghan court convicts US trio of torture KABUL, Afghanistan – Three Americans – led by a former Green Beret who boasted he had Pentagon support – were found guilty yesterday of torturing Afghans in a private jail and were sentenced to prison. 5. Soldier to Plead Guilty in Iraq Abuse Case (AP) AP - An Army reservist charged with abusing Iraqi prisoners plans to plead guilty at a court martial to four counts arising from the Abu Ghraib prison abuse scandal in a plea deal in which eight other counts will be dropped, his lawyer has said.
(DBpedia) Neuron #71: the artist's born date.	1. Joanna Taylor (born 24 July 1978) is an English actress and former model. 2. Jody Miller (born November 29 1941) is an American country music singer. Born as Myrna Joy Miller she was born in Phoenix Arizona and raised in Oklahoma. 3. Priscilla Mitchell (born September 18 1941 in Marietta Georgia) was an American country music singer. 4. Geoffrey Davies (born 15 December 1942 Leeds West Riding of Yorkshire) is a British actor. 5. He was born in Asunción Paraguay on March 27 1950. Son of Carmen Emategui and Rodolfo Barreto.

Table 5. The explanations generated by CB-LLM w/ ACC for two text samples.



Sample	Explanations
(SST2) Sample #330: occasionally funny , always very colorful and enjoyably overblown in the traditional almodóvar style .	1. Charming characters. 2. Clever and unexpected humor. 3. Stunning and exotic locations. 4. Stellar and diverse ensemble cast. 5. Unique and well-developed characters.
(YelpP) Sample #34857: This place has something for everyone. My wife and I started going there out of convenience before attending a movie at the South Pointe. But then we continued going back because we liked the food and the staff is very helpful. This most recent visit I had sushi for the first time and it was very good - and reasonably priced. We have company coming and are going to make it one of our stops on their visit.	1. Welcoming and friendly staff. 2. Clean and inviting ambiance. 3. Amazing flavors. 4. Great warranty and support. 5. Delicious food.

Tables regarding interpretability



Unlearned concept: Overpriced

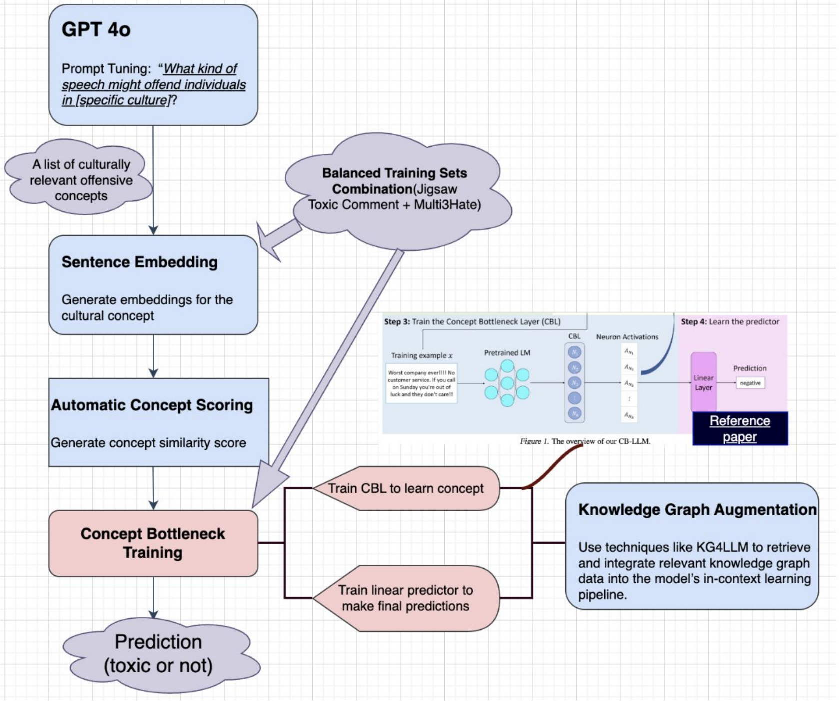
Example:

I was not impressed but was shocked by the prices. RIDICULOUS! I guess they assume you have extra money to throw around since you're in LV. Anyhow, loved the lobster tails besides the excessive amount of filling.

Figure 6. An example of concept unlearning. This example is initially classified as negative due to the customer complaining about the high price, despite the lobster tails being great. After unlearning the concept "Overpriced", the concepts "Amazing flavors" and "Generous portion sizes" dominate the prediction, resulting in a positive prediction.

Future Work

Extend the use of CB-LLM to ***toxic speech detection****. Enhance explainability in* ***cross-cultural, multilingual*** *tasks, which is crucial for reducing bias and improving fairness in NLP systems.*



GPT 4o

Prompt Tuning: "*What kind of speech might offend individuals in [specific culture]*?

A list of culturally relevant offensive concepts

Balanced Training Sets Combination(Jigsaw Toxic Comment + Multi3Hate)

Sentence Embedding

Generate embeddings for the cultural concept

Automatic Concept Scoring

Generate concept similarity score

Concept Bottleneck Training

Train CBL to learn concept

Train linear predictor to make final predictions

Knowledge Graph Augmentation

Use techniques like KG4LLM to retrieve and integrate relevant knowledge graph data into the model's in-context learning pipeline.

Prediction (toxic or not)

Step 3: Train the Concept Bottleneck Layer (CBL)
Training example x
Worst company ever!!!! No customer service. If you call on Sunday you're out of luck and they don't care!!
Pretrained LM
CBL
N_1
N_2
N_3
N_k
Neuron Activations
A_{N_1}
A_{N_2}
A_{N_3}
A_{N_k}
Step 4: Learn the predictor
Linear Layer
Prediction
negative
Reference paper

Figure 1. The overview of our CB-LLM.

My Idea

Existing **toxic speech detection models** have several limitations:

1. They only generate a **toxicity score** without explaining **why** a statement is toxic.
2. When moderators **disagree** with the model's judgment, they **cannot manually adjust or correct the decision**.
3. Especially in **cross-cultural contexts**, where the same statement might be offensive in culture A but acceptable in culture B, making accurate judgment even more difficult.

To address these issues, I would like to apply **CB-LLM** to toxic speech detection. To make it culturally adaptive:

- **Generate key concepts** (e.g., "insult," "sarcasm," "offense").
- **Apply ACS+ACC** on high-quality, multilingual datasets to detect culture-sensitive toxic concepts.
- **Adjust concept weights** to adapt to different cultural backgrounds.

Datasets

I plan to use **large-scale, high-quality, multilingual** datasets that contain diverse toxic speech examples.

- PolygloToxicityPrompts
    a. **Description**: A multilingual toxicity evaluation benchmark curated from web text.
    b. **Size:** 425,000 prompts labeled for toxicity.
    c. **Language Covered**: 17 languages
- Jigsaw Toxic Comment Dataset
    a. **Description**: dataset for a Kaggle competition, containing comments from Wikipedia talk pages.
    b. **Size**: Over 223,000 comments labeled for toxicity.
    c. **Language Covered**: many different languages(haven't checked), but mainly English
- MMHS150k
    a. **Description**: a multimodal dataset comprising tweets, each containing both text and an associated image.
    b. **Size**: 150,000 tweets.
    c. **Language Covered**: Mainly English

Question: How to address the dataset imbalance problem? Most of the datasets primarily contain English texts.

Evaluation Metrics

Effectiveness

Compute F1 Score and accuracy

Cultural Adaptability

Question: *how to evaluate if the model adapts across different languages and cultures? If our model found out that the same text in language A is not toxic but becomes toxic when it is translated to language B, how can I check it is correct? Human Evaluation?*

Interpretability

Human evaluation: Measure alignment between ACS-generated explanations and human annotations through questionnaire.

Milestones and Timeline

Milestone	Expected Timeline
Data Preparation and Preprocessing	1-2 weeks
CB-LLM Model Training(backbone GPT-2)	3 - 5 weeks
Compare the performance w/ ACS and ACC	
Backbone Model Distillation(if possible)	2 - 4 weeks
Evaluation and Testing	1 week

Questions

1. Are interpretable models like CB-LLM used only for training and testing step, or do they have potential for commercial deployment in apps? How are they typically applied in real-world scenarios?
2. Will deploying CB-LLM on the edge (e.g., integrating it into an app for toxic speech detection) pose a challenge due to its large model size? If so, can distillation techniques be applied to interpretable models such as CB-LLM? Will single-neuron interpretability degrade significantly after distillation?
3. If I want to adjust toxicity weights across different cultures, do I have to create a separate weight table for each culture and manually adjust it just like concept unlearning? Or is there a way to automatically learn/select appropriate weights based on a user's language and IP?

Classification with Conceptual Safeguards

ICLR 2024

Authors: Hailey Joren, Charles Marx, Berk Ustun

UC San Diego

Main Problem

Automation in classification tasks such as medical diagnosis, image recognition, and toxicity detection often faces **reliability concerns**, where even high-accuracy models may require **human oversight** to avoid critical errors.

abstention mechanism

Conceptual safeguards

Allows a model to abstain from predictions when uncertainty is high and involve human experts to confirm intermediate concepts.

Goal: Improve the trade-off between *accuracy* and *coverage*

$$\begin{aligned} \max_{h} \quad & \text{Coverage}(h) \\ \text{s.t.} \quad & \text{Accuracy}(h) \geq \alpha, \end{aligned}$$

Existing Approaches to Solve This Problem

Selective Classification

Traditional selective classifiers use a confidence threshold to decide whether to abstain from predictions. While this improves accuracy, it often sacrifices coverage and does not explicitly integrate human verification.

Deep Learning with Concepts

Concept bottleneck models predict outcomes via intermediate concepts, which can be reviewed or corrected by humans. However, these models face challenges with incomplete concept annotations and lack mechanisms for selective abstention.

Key Contributions

1

Conceptual Bottleneck Model(CBM)

- **Concept Detector (g(x))**: Predict intermediate concepts, e.g. "pigmentation" in skin cancer diagnosis.
- **Front-End Classifier (f(c))**: Performs the final classification based on the intermediate concepts.

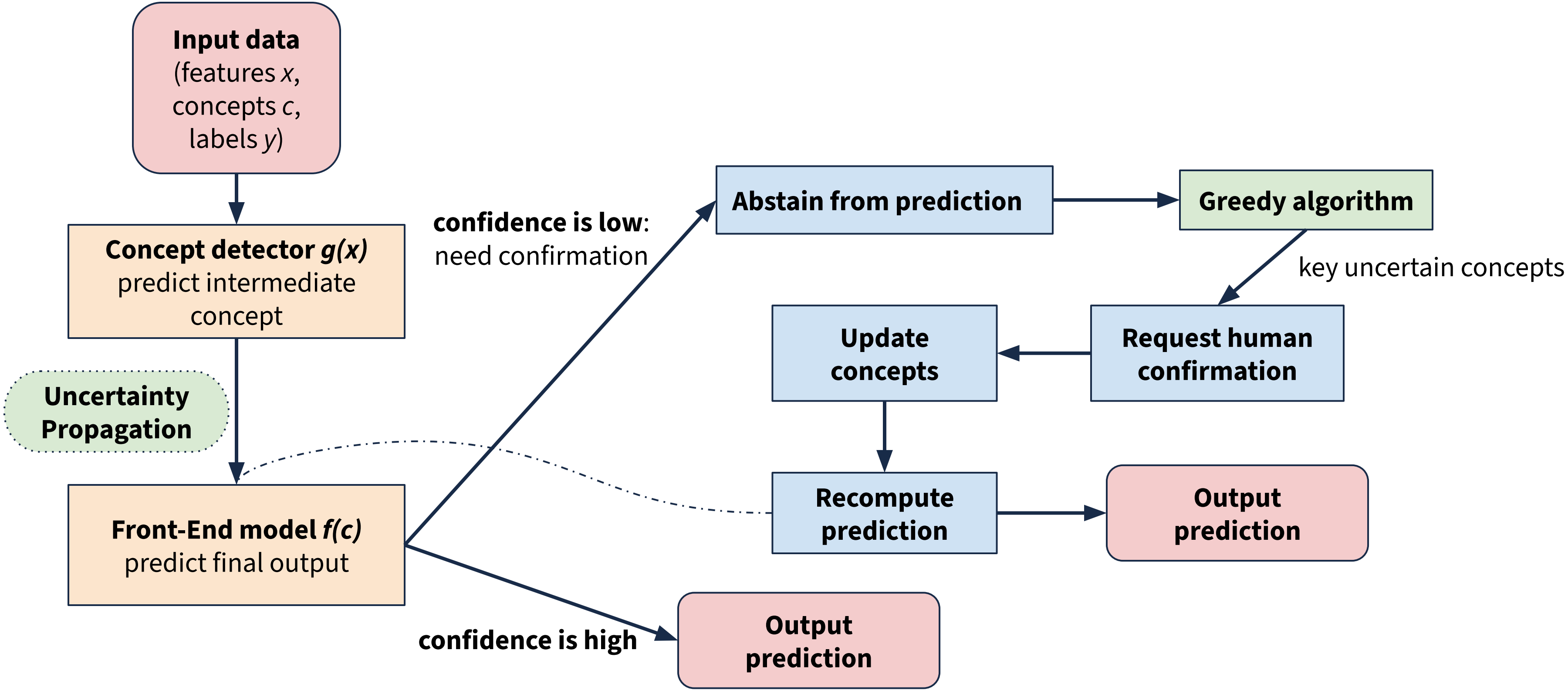
2

Uncertainty Propagation: Enables the front-end model to accept probabilistic concept predictions rather than hard concepts as input, using conditional independence assumptions to propagate uncertainty.

3

Confirmation Policy: A greedy algorithm prioritizes confirming concepts that maximize the reduction in uncertainty within a given budget.

Workflow



Input data (features *x*, concepts *c*, labels *y*)

Concept detector *g(x)* predict intermediate concept

Uncertainty Propagation

Front-End model *f(c)* predict final output

confidence is low: need confirmation

Abstain from prediction

Greedy algorithm

key uncertain concepts

Request human confirmation

Update concepts

Recompute prediction

Output prediction

confidence is high

Output prediction

Framework

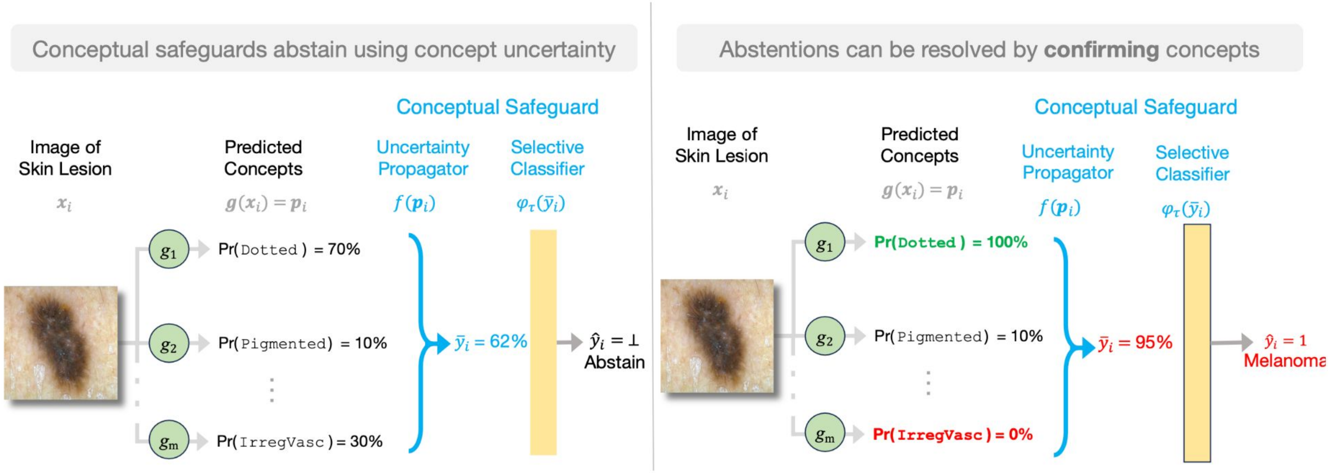
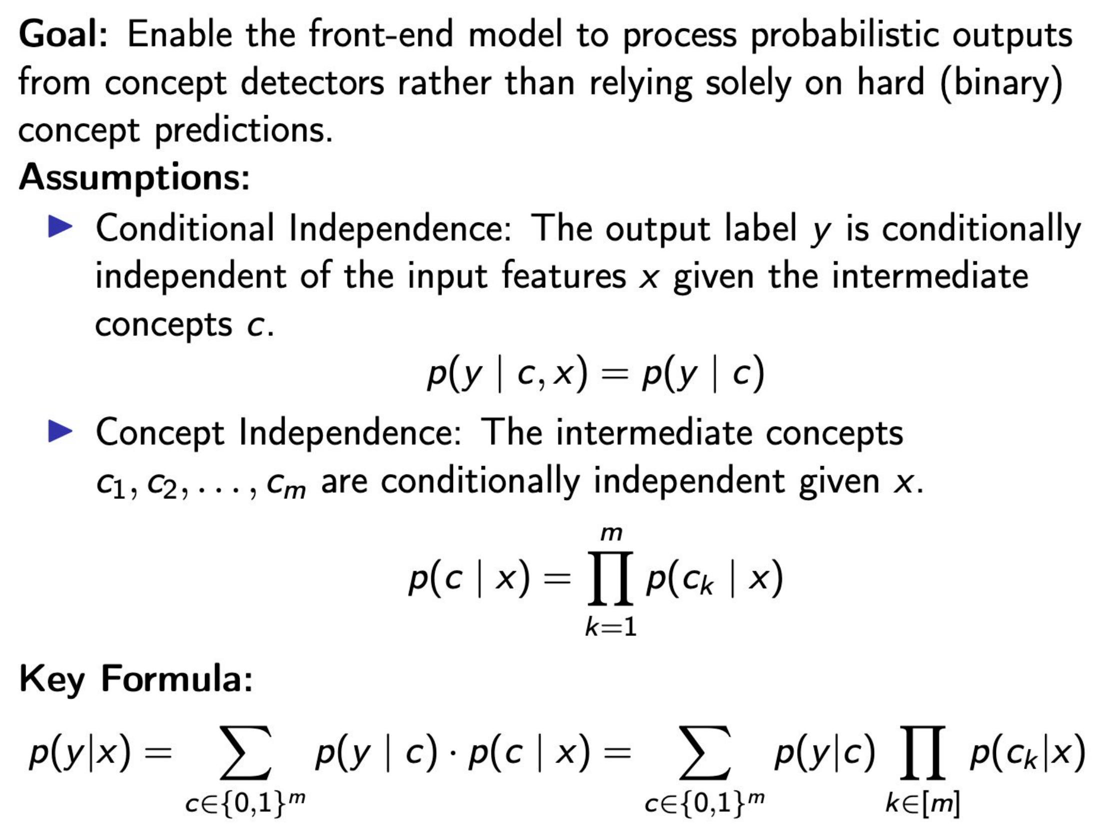


Figure 1: Conceptual safeguard to detect melanoma from an image of a skin lesion. We consider a model that estimates the probabilities of m concepts: `Dotted`, `Pigmented` . . . `IrregularVasc`. Given these probabilities, a conceptual safeguard will decide whether to output a prediction $\hat{y} \in \{\texttt{Melanoma}, \texttt{NoMelanoma}\}$ or to abstain $\hat{y} = \perp$. The safeguard improves accuracy by abstaining on images that would receive a low confidence prediction, and measures confidence in a way that accounts for the uncertainty in concept predictions through uncertainty propagation. On the left, we show an image where a safeguard abstains because its confidence fails to meet the threshold to ensure high accuracy $\Pr(\texttt{Melanoma}) = 62\% \leq 90\%$. On the right, we show a human expert can resolve the abstention by confirming the presence of concepts `Dotted` and `IrregVasc` in the image.

Key Equations: Uncertainty Propagation



Goal: Enable the front-end model to process probabilistic outputs from concept detectors rather than relying solely on hard (binary) concept predictions.

Assumptions:

- Conditional Independence: The output label y is conditionally independent of the input features x given the intermediate concepts c.

$$p(y \mid c, x) = p(y \mid c)$$

- Concept Independence: The intermediate concepts $c_1, c_2, \dots, c_m$ are conditionally independent given x.

$$p(c \mid x) = \prod_{k=1}^{m} p(c_k \mid x)$$

Key Formula:

$$p(y|x) = \sum_{c \in \{0,1\}^m} p(y \mid c) \cdot p(c \mid x) = \sum_{c \in \{0,1\}^m} p(y|c) \prod_{k \in [m]} p(c_k|x)$$

Key Equations: Uncertainty Propagation

Key Points:

- $p(c_k \mid x)$ is the probability of concept c_k given x, computed by the concept detector.
- $p(y \mid c)$ is the probability of the label y given the intermediate concepts c, computed by the front-end model.

Binary Concept Representation:

For each concept $c_k \in \{0, 1\}$:

- $p(c_k \mid x) = q_k$, where q_k is the probability that $c_k = 1$.
- $1 - q_k$ is the probability that $c_k = 0$.

Joint Probability:

$$p(c \mid x) = \prod_{k=1}^{m} q_k^{c_k}(1 - q_k)^{1-c_k}$$

Key Equations: Uncertainty Propagation

The final expression for $p(y \mid x)$ becomes:

$$f(\boldsymbol{p}_i) := \sum_{\boldsymbol{c} \in \{0,1\}^m} f(\boldsymbol{c}) \prod_{k \in [m]} p_{i,k}^{c_k} (1 - p_{i,k})^{1-c_k}$$

$$\underbrace{f(\boldsymbol{p}_i)}_{\substack{\text{output using} \\ \text{uncertain concepts}}} := \sum_{\mathbf{c} \in \{0,1\}^m} \underbrace{p(y \mid \boldsymbol{c})}_{\substack{\text{output using hard} \\ \text{concepts}}} \prod_{k \in [m]} \underbrace{p(c_k \mid \boldsymbol{x})}_{\substack{\text{output from concept} \\ \text{detectors}}}$$

Efficient Approximation

- For a small number of concepts (m), compute the full sum over 2^m combinations of c.
- For a large m, use Monte Carlo sampling to approximate the sum.

Key Equations: Uncertainty Propagation

The final expression for $p(y \mid x)$ becomes:

$$f(\boldsymbol{p}_i) := \sum_{\boldsymbol{c}\in\{0,1\}^m} f(\boldsymbol{c}) \prod_{k\in[m]} p_{i,k}^{c_k}(1 - p_{i,k})^{1-c_k}$$

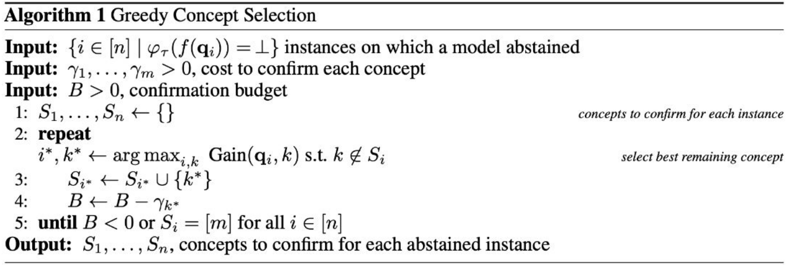
$$\underbrace{f(\boldsymbol{p}_i)}_{\substack{\text{output using}\\ \text{uncertain concepts}}} := \sum_{\mathbf{c}\in\{0,1\}^m} \underbrace{p(y \mid \boldsymbol{c})}_{\substack{\text{output using hard}\\ \text{concepts}}} \prod_{k\in[m]} \underbrace{p(c_k \mid \boldsymbol{x})}_{\substack{\text{output from concept}\\ \text{detectors}}}$$

Efficient Approximation

- For a small number of concepts (m), compute the full sum over 2^m combinations of c.
- For a large m, use Monte Carlo sampling to approximate the sum.

Key Algorithm: Greedy concept selection

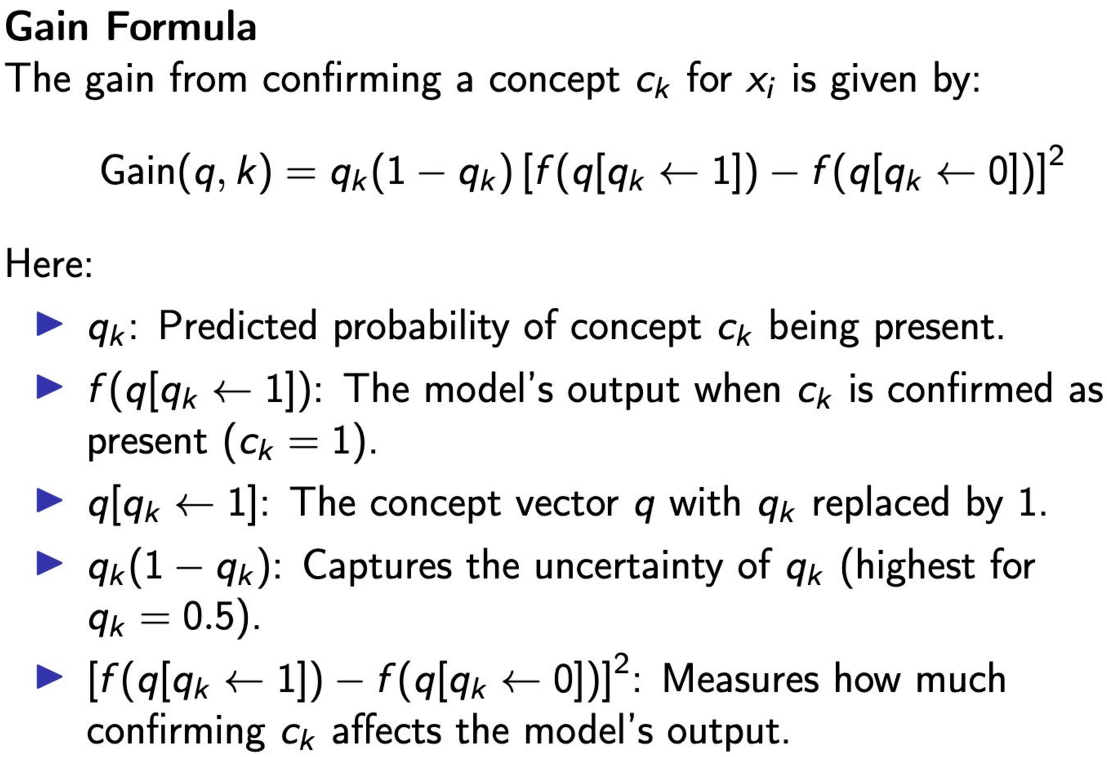
Algorithm 1 is designed to prioritize confirming concepts for instances where the model abstains. It selects the most promising concept to confirm based on a calculated gain metric, within a given budget.



```
Algorithm 1 Greedy Concept Selection
Input: $\{i \in [n] \mid \varphi_\tau(f(\mathbf{q}_i)) = \perp\}$ instances on which a model abstained
Input: $\gamma_1, \ldots, \gamma_m > 0$, cost to confirm each concept
Input: $B > 0$, confirmation budget
$S_1, \ldots, S_n \leftarrow \{\}$                                    concepts to confirm for each instance
repeat
$i^*, k^* \leftarrow \arg\max_{i,k}$ Gain$(\mathbf{q}_i, k)$ s.t. $k \notin S_i$      select best remaining concept
    $S_{i^*} \leftarrow S_{i^*} \cup \{k^*\}$
    $B \leftarrow B - \gamma_{k^*}$
until $B < 0$ or $S_i = [m]$ for all $i \in [n]$
Output: $S_1, \ldots, S_n$, concepts to confirm for each abstained instance
```

1. Identifies the concept with the highest expected gain in reducing uncertainty.
2. Updates the confirmation set for the chosen instance and adjusts the remaining budget.
3. Repeats until the budget is exhausted or all concepts have been confirmed.

Key Algorithm: Greedy concept selection



Gain Formula

The gain from confirming a concept c_k for x_i is given by:

$$\text{Gain}(q, k) = q_k(1 - q_k)\left[f(q[q_k \leftarrow 1]) - f(q[q_k \leftarrow 0])\right]^2$$

Here:

- q_k: Predicted probability of concept c_k being present.
- $f(q[q_k \leftarrow 1])$: The model's output when c_k is confirmed as present $(c_k = 1)$.
- $q[q_k \leftarrow 1]$: The concept vector q with q_k replaced by 1.
- $q_k(1 - q_k)$: Captures the uncertainty of q_k (highest for $q_k = 0.5$).
- $\left[f(q[q_k \leftarrow 1]) - f(q[q_k \leftarrow 0])\right]^2$: Measures how much confirming c_k affects the model's output.

Experiment Setup

- **Datasets**: melanoma and skincancer from Derm7pt datasets to diagnose diseases, warbler and flycatcher to identify birds, noisyconcepts25 and noisyconcepts75 to control the noise.

- **Models**:
  - *X→Y MLP*: A multilayer perceptron trained on top of the penultimate layer of the embedding model. This baseline represents an end-to-end deep learning model that directly predicts the output without concepts.
  - *Baseline*: An independent concept bottleneck model built from concept detectors and a front-end model trained to predict the true concepts.
  - *CS*: Conceptual safeguard built from the same concept detectors and front-end as the baseline.

- **Comparison Metrics:**
  - accuracy and coverage trade-offs
  - effect of uncertainty propagation and confirmation through ablation studies

Experiment Result

Accuracy-Coverage Trade-off:
Overall, CS outperformed baseline models, achieving a better balance between accuracy and coverage.

On Uncertainty Propagation:

- Uncertainty propagation improves the accuracy coverage trade-off
- Accounting for uncertainty can improve these trade-offs by producing a more effective confirmation policy. Gains of uncertainty may change under a confirmation budget.

On Confirmation Policies:
Confirming concepts improves performance across all datasets, especially under limited confirmation budgets.

Future Work

- ❖ CS may improve safety and fairness of toxic speech detection by incorporating human review for culturally sensitive or ambiguous concepts.

- ❖ The framework’s focus on uncertainty propagation and human confirmation aligns with the goal of enhancing model transparency in multilingual toxic speech detection.

Thank You!